

Local Distributed Algorithms for Selfish Agents*

Simon Collet^{†1}, Pierre Fraigniaud¹, and Paolo Penna²

¹CNRS and University Paris Diderot, France.

²ETH Zurich, Switzerland.

Abstract

In the classical framework of *local distributed network computing*, it is generally assumed that the entities executing distributed algorithms are *altruistic*. However, in various scenarios, the value of the output produced by an entity may have a tremendous impact on its future. This is for instance the case of tasks such as computing maximal independent sets (MIS) in networks. Indeed, a node belonging to the MIS may be later asked more than to a node not in the MIS — e.g., because MIS in networks are often used as backbones to collect, transfer, and broadcast information, which is costly. In this paper, we revisit typical local distributed network computing tasks in the framework of algorithmic game theory. Specifically, we focus on the construction of solutions for *locally checkable labeling* (LCL) tasks, which form a large class of distributed tasks, including MIS, coloring, maximal matching, etc., and which have been studied for more than 20 years in distributed computing.

Given an LCL task, the nodes are collectively aiming at computing a solution, but, at the individual level, every node plays *rationaly* and *selfishly* with the objective of optimizing its own profit. Surprisingly, the classical frameworks for game theory are not fully appropriate for the purpose of our study. Moreover, we show that classical notions like *Nash equilibria* may yield algorithms requiring an arbitrarily large number of rounds to converge. Nevertheless, by extending and generalizing core results from game theory, we establish the existence of a so-called *trembling-hand perfect equilibria*, a subset of Nash equilibria that is well suited to LCL construction tasks. The main outcome of the paper is therefore that, for essentially all distributed tasks whose solutions are locally checkable, there exist construction algorithms which are robust to selfishness, i.e., which are guaranteeing that nodes have no incentive to deviate from the actions instructed by these algorithms.

*All authors received supports from the ANR project DISPLEXITY, and from the INRIA project GANG.

[†]Funded by the European Research Council (ERC) under the H2020 research and innovation program (grant No 648032).

1 Introduction

1.1 Context and Objective

In the conceptual framework of local distributed computing in networks [27], algorithms solving tasks such as minimum-weight spanning tree, maximum matching, coloring, maximal independent set, etc., usually assume that every node of the network will obey their instructions. This might be appropriate in a context in which each node is altruistic and/or oblivious to its output. However, there exist several settings in which the future of a node depends on its output. For instance, independent sets and dominating sets can be used as backbones to collect, transfer, and broadcast information, and/or as cluster heads in clustering protocols [16, 20]. Hence, belonging to the independent set or to the dominating set may generate future costs in term of energy consumption, computational efforts, and bandwidth usage. As a consequence, rational selfish nodes might be tempted to deviate from the instructions of the algorithm, so that to avoid becoming member of the independent set or dominating set under construction. On the other hand, the absence of a backbone or of cluster heads may penalize the nodes. Hence every node is subject to a tension between (1) facilitating the construction of a solution, and (2) avoiding certain types of solutions. The objective of this paper is to design algorithms enabling to resolve this tension.

For example, a large class of randomized maximal independent set (MIS) algorithms [1, 19] proceed in rounds, where a round allows every node to exchange information with its neighbors in the network. Roughly, at each round, every node i which has not yet decided proposes to enter the MIS with a certain probability p_i . If a node proposes itself to enter the MIS, and none of its neighbors is also proposing itself to enter the MIS, then it enters the MIS. If two adjacent nodes simultaneously propose themselves to enter the MIS, then there is a conflict, and both nodes remain undecided. The quality of the algorithm heavily depends on the choice of the probability p_i that node i proposes itself to enter the MIS, which may typically depend on the degree of node i , and may vary along with the execution of the algorithm as node i accumulates more and more information about its neighborhood. Hence, a node i aiming at avoiding entering the MIS might be tempted to deviate from the algorithm by setting p_i small. However, if all nodes deviate in this way, then the algorithm may take a very long time before converging to a solution.

Coloring is another example where nodes may adopt selfish strategies to avoid some specific solutions. This is for instance the case where coloring is viewed as an abstraction of frequency assignment in radio networks. Some frequencies might be preferred to others because, e.g., some frequencies might be conflicting with local devices present at the node. As a consequence, not all colors are equal for the nodes, and while each node is aiming at constructing a coloring quickly (in order to take benefit from the resulting radio network), it is also aiming at avoiding being assigned a color that does not please it. Now, similarly to MIS construction, a large class of randomized coloring algorithms proceed in rounds, roughly as follows (see, e.g., [3]). At each round, every node that has not yet decided a color (1) picks a color uniformly at random among the available colors (i.e., among a given palette, but the colors already decided by neighboring nodes), (2) checks whether this color is conflicting with the random colors picked by the neighbors, and (3) decides this color in absence of conflicts. If a node has preferences to some colors, then it might be tempted to give more weight to its preferred colors in its random choices. Again, if all nodes deviate this way, then the algorithm may take a long time before converging to a solution, if converging at all.

In this paper, we address the issue of designing distributed algorithms that are robust to selfish behaviors of the nodes. Obviously, algorithmic game theory has been conceived especially for

tackling such issues. So, more precisely, we are aiming at answering the following question:

Existence of algorithms coping with non-cooperative behavior: Do there exist some forms of equilibria in the games formalizing distributed tasks such as the one sketched above? That is, do there exist strategies, i.e., probability distributions over the random choices to be made by each node at each round, from which no rational selfish players have incentive to deviate? More importantly, if such equilibria exist, are they corresponding to efficient algorithms for solving the tasks?

1.2 Our Results

First, we define *LCL games*, induced by a generic form of randomized algorithms for constructing solutions to *locally checkable labeling* (LCL) tasks (cf. Section 2). Recall that locally checkable labeling tasks [21] form a large class of classical problems, including maximal independent set, coloring, maximal matching, minimal dominating set, etc., studied for more than 20 years in the framework of distributed computing.

Next, we observe that classical frameworks for algorithmic game theory do not fully capture the essence of distributed games (cf. Section 3). Moreover, we show that the classical notion of Nash equilibria is not sufficient to characterize desirable distributed algorithms, i.e., efficient and robust to the selfish behaviors of the nodes. In particular, we show that there are Nash equilibria which correspond to algorithms whose running times, i.e., number of rounds, are arbitrarily large (cf. Proposition 1).

In contrast, we show that the stronger notion of *trembling-hand perfect equilibria* (a subset of Nash equilibria) does not suffer from the above problem. In fact, trembling-hand perfect equilibrium is a notion of equilibrium that is well suited to the purpose of designing efficient and robust randomized distributed algorithms for LCL tasks. Indeed, trembling-hand perfect equilibria are limits of Nash equilibria of perturbed games, that is, games in which players must play completely mixed strategies (i.e. strategies in which each action has a constant probability of being played). As such, trembling-hand perfect equilibria do not suffer from the undesirable phenomena of players blocking each other for an arbitrarily long time. Indeed, this phenomenon results from weakly dominated strategies that stick to Nash equilibria in which players repeat the same choices again and again for an arbitrarily long amount of time.

We prove that all LCL games whose solution can be constructed by a greedy sequential algorithm, such as MIS, $(\Delta + 1)$ -coloring, maximal matching, etc., have trembling-hand perfect equilibria. Specifically, our main result is the following:

Theorem 1 *Let \mathcal{L} be a greedily constructible locally checkable labeling. The game associated to \mathcal{L} has a symmetric trembling-hand perfect equilibrium.*

This result is achieved by establishing a general result, of independent interest, stated below as a lemma:

Lemma 1 *Every infinite, continuous, measurable, well-rounded, symmetric extensive game with perfect recall and finite action set has a symmetric trembling-hand perfect equilibrium.*

Our results open a new line of research aiming at designing distributed algorithms that are not only efficient, but also robust to selfish behaviors of the nodes. Indeed, Theorem 1 establishes that such algorithms *exist*, for all “natural” locally checkable labeling (LCL) tasks.

1.3 Additional Related Work

The readers interested in specific important distributed network computing problems such as MIS and $(\Delta + 1)$ -coloring are invited to consult [9, 12, 15] and the references therein for the most recent advances in the field. See also [31] for a survey, and [5, 8, 21] for results on derandomization techniques for LCL tasks.

Other forms of game theoretic approaches have been studied in the past, in the context of distributed computing, and/or using a distributed computing approach for solving graph problems. In [6], the authors study programming incentives to the programmers for writing codes that are efficient for the overall system while optimizing individual programs performances, in the context of transactional memory. In [2], the authors describe and analyze ways of computing max-cuts by interpreting these cuts as the pure Nash equilibria of an n -player non-zero sum game, each vertex being an agent trying to maximize her selfish interest.

The papers the most closely related to our work are [23, 24, 25]. In the survey [23], the author addresses several problems related to distributed computing, under a large spectrum of different models. It is shown that standard mechanism design techniques (e.g., Vickrey-Groves-Clark mechanism) can be applied to some of these problems. However, such an application requires a central authority which knows everything about the system, except of course the players private information. On the other hand, the paper also describes and analyzes a two-step algorithm for solving MIS on paths, where the players are bounded to communicate only with their neighbors. The algorithm however requires a trusted entity that distributes payments to the players. The class of best-response mechanisms is studied in [25]. Such mechanisms apply to strategic games with a unique equilibrium, in which the players proceed by repeatedly best-respond to each other. Several important problems can be casted in this framework, including the BGP protocol, TCP-like games, among others. However, for applying the mechanism, the underlying game must have a unique equilibrium (plus other technical conditions), and the players are oblivious about the time required to compute the solution. Even the first condition alone may not be satisfied by LCL games (it is typically not satisfied by the MIS game).

In contrast to the above, our setting does not assume any central entity nor payments, it applies to general graphs and to all “natural” LCL tasks (including MIS). We prove the existence of suited algorithms in this general setting.

2 LCL Games

2.1 LCL languages

Our framework is distributed network computing, for which we adopt the following standard setting (see, e.g., [10]). Let A be a finite alphabet. A network *configuration* is a pair (G, s) where $G = (V, E)$ is a simple connected graph, and $s : V \rightarrow A$ is a state function, assigning a state $s(v) \in A$ to every node $v \in V$. A *distributed language* over a graph family \mathcal{F} is a Turing computable set \mathcal{L} of configurations such that, for every graph $G \in \mathcal{F}$, there is a state function s for which $(G, s) \in \mathcal{L}$. The class LCL, for *locally checkable labeling*, has been defined in [21]. Let $\Delta \geq 2$, and let \mathcal{F}_Δ be the family of simple connected graphs with maximum degree Δ . An LCL language over \mathcal{F}_Δ is a distributed language defined by a set of balls of radius at most t in graphs belonging to \mathcal{F}_Δ , for some $t \geq 0$, with nodes labeled by values in A . These balls are called the *good* balls of \mathcal{L} . The set of good balls of an LCL language \mathcal{L} is denoted by $\text{good}(\mathcal{L})$. The parameter t is called the *radius*

of \mathcal{L} . Throughout the paper, we shall solely consider graphs with degrees upper bounded by some $\Delta \geq 2$, and thus we ignore the index Δ in \mathcal{F}_Δ in the following.

For instance, MIS is the LCL language of radius 1 with $A = \{0, 1\}$, where $\text{good}(\mathcal{L})$ is the set of balls with radius 1 centered at a node v , such that either $s(v) = 0$ and there exists a neighbor w of v with $s(w) = 1$, or $s(v) = 1$ and $s(w) = 0$ for every neighbor w of v . Similarly, $(\Delta + 1)$ -coloring is the LCL language of radius 1 with $A = \{1, \dots, \Delta + 1\}$, where $\text{good}(\mathcal{L})$ is the set of balls with radius 1 whose center v satisfies $s(v) \neq s(w)$ for every neighbor w of v .

Every LCL language \mathcal{L} induces an associated construction *task*. Given $G \in \mathcal{F}$, each of its nodes v has to compute a value $s(v) \in A$ such that $(G, s) \in \mathcal{L}$. In order to guarantee the existence of efficient distributed algorithms for solving the task induced by \mathcal{L} , we consider a subclass of LCL languages, called *greedily constructible*. Roughly, an LCL language \mathcal{L} is greedily constructible if any partial solution can be extended. More formally, let us consider an extra value \perp , whose interpretation is that a node v with $s(v) = \perp$ has not yet decided an output value in A . For a language \mathcal{L} of radius t , a radius- t ball B is said to be *partially good* if B is resulting from a good ball by replacing the value of some of its node labels by \perp . Let $G = (V, E) \in \mathcal{F}$, and let $s : V \rightarrow A \cup \{\perp\}$ for which all balls in (G, s) are partially good. Let $U = \{v \in V : s(v) = \perp\}$, and let $s' : V \rightarrow A$ be such that $s'(v) = s(v)$ for all nodes $v \in V \setminus U$. We call such a function s' an *extension* of s . Let U' be composed of all nodes in U whose balls in (G, s') are good, and let $s'' : V \rightarrow A \cup \{\perp\}$ be defined as $s''(v) = s(v)$ for $v \in V \setminus U$, $s''(v) = \perp$ for $v \in U \setminus U'$, and $s''(v) = s'(v)$ for $v \in U'$. An LCL language \mathcal{L} is greedily constructible if for every such G and s such that $U \neq \emptyset$, we have (1) there exists an extension s' of s such that $s'' \neq s$, and, (2) for every extension s' of s such that $s'' \neq s$, all balls in (G, s'') are partially good. For instance, k -coloring is greedily constructible for any $k > \Delta$, while Δ -coloring is not (even in Δ -colorable graphs). MIS and maximal matching also fall into this class as long as we enforce that nodes (resp., edges) cannot decide by themselves not to enter the MIS (resp., matching), but can only do so *after* a neighbor has entered the MIS (resp., matching).

2.2 A generic randomized algorithm for LCL languages

The generic randomized construction algorithm for LCL languages described in this subsection is conceived for the classical LOCAL model of distributed computation [27]. Recall that this model assumes that all nodes start simultaneously, and perform synchronously in a sequence of rounds. At each round, every node exchanges information with its neighbors, and performs individual computations. This goes on at every node until the node decides some output value, and terminates. The nodes do not need to terminate at the same round. However, the output value of a terminated node is irrevocable. Our generic algorithm is described in Algorithm 1.

Algorithm 1 *Generic algorithm executed at a node v .*

```

1:  $\text{val}(v) \leftarrow \perp$ 
2: observe  $\text{ball}(v)$ 
3: while  $\text{ball}(v) \notin \text{good}(\mathcal{L})$  do
4:   set  $\text{val}(v) \in A$  compatible with  $\text{ball}(v)$ , at random, according to a probability distribution  $\mathcal{D}$ 
5:   observe  $\text{ball}(v)$ 
6: end while
7: return  $\text{val}(v)$ 

```

In Algorithm 1, every node v aims at computing $\text{val}(v) \in A$. It starts with initial value $\text{val}(v) = \perp$. At Step 2 node v collects the t -ball $\text{ball}(v)$ around it, i.e., the values of all the labels of

nodes at distance at most t from it, including the structure of the ball, where t is the radius of \mathcal{L} . (This can be implemented in t rounds in the LOCAL model). At Step 3, node v checks whether the observed ball, $ball(v)$, is good. If not, then Step 4 consists for v to select a random value $val(v) \in A$ compatible with $ball(v)$, i.e., with values of nodes in the ball fixed at previous rounds. The random choice of $val(v)$ is governed by a probability distribution \mathcal{D} , which may depend on the round, and on the structure of v 's neighborhood as observed during the previous rounds. In the case of MIS, this choice essentially consists in choosing whether v proposes itself for entering the MIS, with probability p_v . Luby's algorithm [19] sets $p_v = \frac{1}{2 \deg(v)}$ where $\deg(v)$ is the degree of v in the graph, not counting the neighbors which already decided at previous rounds. Similarly, in the case of $(\Delta + 1)$ -coloring, the choice of $val(v)$ essentially consists in proposing a color at random. Barenboim-Elkin's algorithm [3] sets the probability p_c of proposing color c as $\frac{1}{2}$ for $c = 0$ (a "fake" color), and $p_c = \frac{1}{2k}$ for any of the k colors $c \in \{1, \dots, \Delta + 1\}$ that have not yet been decided by some neighboring node(s) during the previous rounds. Once the new value $val(v)$ has been selected, node v observes $ball(v)$ in Step 5, including all selected values of nodes in this ball. If the selected values by the nodes in this ball make the ball good, then v terminates and outputs $val(v)$. Otherwise, v starts a new iteration of the while-loop.

For any greedily constructible language \mathcal{L} , the algorithm terminates (i.e., all nodes return a value) as long as every value in A has non zero probability to be chosen, and the collective output $val = \{val(v), v \in V\}$ is such that $(G, val) \in \mathcal{L}$ for any graph $G = (V, E)$. The performances of the algorithm (i.e., how many rounds it takes until all nodes terminate) however depends on the choice of the probability distributions \mathcal{D} chosen by each node $v \in V$ at each round. For instance, in the case of MIS, the algorithm converges if every node v proposes itself to enter the MIS with probability $p_v = \frac{1}{2}$. However, the choice $p_v = \frac{1}{2 \deg(v)}$ as in Luby's algorithm guaranties a much faster convergence: $O(\log n)$ rounds, w.h.p., in n -node networks.

2.3 A distributed game for LCL languages

Let \mathcal{L} be a greedily constructible LCL language of radius t . \mathcal{L} is associated to a game, that we call LCL game, and that we define as follows (for a more formal definition, see Section 5.1). Let G be a connected simple graph. Every node v of G is a rational and selfish *player*, playing the game with the ultimate goal of maximizing its payoff. Roughly, a *strategy* for a node v is a probability distribution \mathcal{D} over the values in A compatible with the t -ball centered at v such as observed by v , employed at Step 4 in Algorithm 1, which may depend on the history of v during the execution of the algorithm.

As mention in the introduction, we want the payoff of the nodes to reflect the tension between the objective of every node to compute a global solution rapidly (for this global solution brings benefits to every node), and its objective of avoiding certain forms of solutions (which may not be desirable from an individual perspective). So, to define the payoff of a node, we denote by *pref* a *preference function*, which is an abstraction of how much a node will "suffer" in the future according to a computed solution. For instance, in the MIS game, for a computed MIS I , we could have, e.g., $pref(I) = 1$ from a node not in I , and $pref(I) = 0$ for a node in I . In general, for $G = (V, E)$, we define $pref: good(\mathcal{L}) \rightarrow \mathbb{R}^+$ by associating to each ball $B \in good(\mathcal{L})$, the preference $pref(B)$ of the center of ball B . The *payoff function* π_v of node v at the completion of the algorithm is decaying with the number k of iterations of the while-loop in Algorithm 1 before the algorithm terminates. More precisely, we set $\pi_v = \delta^k pref(B_v)$ where $0 < \delta < 1$, B_v is the t -ball centered at v returned by Algorithm 1, and k is the number of while-loop iterations performed by Algorithm 1 before all

nodes in B_v output their values. The choice of $\delta \in (0, 1)$ reflects the tradeoff between the quality of the solution from the nodes' perspective, and their desire to construct a global solution. Note that k is at least the time it took for v to decide a value in A , and at most the time it took for the algorithm to terminate at all nodes. If the algorithm does not terminate around v , that is, if the t -ball around v remains perpetually undecided in at least one of its nodes, then we set $\pi_v = 0$.

The payoff of a node v will thus be large if all nodes in its t -ball decide quickly (i.e., k is small), and if the computed t -ball B_v suits node v (i.e., $\text{pref}(B_v)$ is large). Conversely, if v or another node in its t -ball takes many rounds before deciding a value (i.e., k is large), or if node v is not satisfied by the computed solution B_v (i.e., $\text{pref}(B_v)$ is small), then the payoff of v will be small. In particular, if $\text{pref}(B) = \text{pref}(B')$ for every good balls B and B' , then maximizing the payoff is equivalent to completing the task quickly. Instead, if $\text{pref}(B)$ is very small for some good balls B , then nodes might be willing to slow down the completion of the task, with the objective of avoiding being the center of these balls, in order to maximize their payoff. That is, such nodes may bias their distribution \mathcal{D} towards preferred good balls, even if this is to the price of increasing the probability of conflicting with the choices of neighboring nodes, resulting in more iterations for reaching convergence.

As mentioned before, a *strategy* of a node at one iteration of Algorithm 1 is a distribution of probability over the values in A , which may depend on the history of the node at any point in time during the execution of Algorithm 1. Note that, at the k th iteration of the while-loop, every node v has acquired knowledge about its environment at distance kt in the actual graph, where t is the radius of the considered language. Importantly, the strategies depend only on the knowledge accumulated by the nodes along the execution of the algorithm. In particular, the fact that nodes might be given distinct identities, as in the LOCAL model, is ignored by the nodes to set their strategies, but used only for distinguishing the nodes in the network, in a way similar to the way classical randomized distributed algorithms perform. In fact, at the beginning of the algorithm, player v does not even know which node she will play in the network, and just knows that the network belong to some given graph family \mathcal{F} .

A first question of interest is whether there a way to assign a strategy profile to the nodes so that no nodes will have incentive to deviate from their given individual strategies? In other words, do there exist Nash equilibria for LCL games?

3 The nature of LCL games

3.1 LCL games as forms of extensive games with imperfect information

Table 1 surveys the existence theorems for equilibria in various game settings, from the finite strategic games to the extensive games with imperfect information. We restrict our attention to games with a finite number of players (for games with infinitely many players, see [26]). Trembling hand equilibria are refinements of sequential equilibria, which are themselves refinements of subgame-perfect equilibria, all of them being Nash equilibria.

In Table 1, we distinguish strategic games (i.e., 1-step games like, e.g., prisoner's dilemma) from extensive games (i.e., game trees with payoffs, like, e.g., monetary policy in economy). For the latter class, we also distinguish games with perfect information (i.e., every player knows exactly what has taken place earlier in the game), from the games with imperfect information. LCL games belongs to the class of extensive games with imperfect information, since a node plays arbitrarily

	Strategic Games	Extensive games with perfect information	Extensive games with imperfect information
Finite games	Nash, 1950 [22] Nash equilibrium	Selten, 1965 [29] Subgame-perfect equilibrium Pure strategies	Selten, 1975 [30] Trembling-hand perfect eq. Behavior strategies
Games with a finite action set	Mixed strategies	Fudenberg, Levine, 83 [11] Subgame-perfect equilibrium Pure strategies	Fudenberg, Levine, 83 [11] Sequential equilibrium Behavior strategies
Games with an infinite action set	Fan, 1952 [7] Glicksberg, 1952 [13] Nash equilibrium Mixed strategies	Harris, 1985 [14] Subgame-perfect equilibrium Pure strategies	Chakrabarti, 1992 [4] Nash equilibrium Behavior strategies

Table 1: A summary of main results about the existence of equilibria

many times, and is not necessarily aware of the actions taken by far away nodes in the network.

In Table 1, we also distinguish finite games (i.e., games with a finite number of pure strategies and finite number of repetitions) from games with infinite horizon (i.e., games which can be repeated infinitely often). The latter class of games is also split into games with finite numbers of actions, and games with infinite set of actions (like, e.g., when fixing the price of a product). LCL games belongs to the class of games with infinite horizon and finite action set: the horizon is infinite since neighboring nodes may perpetually prevent each other from terminating, and the action set A is supposed to be finite.

The previous work most closely related to our framework is [11]. Indeed, as was said before, LCL games are extensive games with imperfect information, and finite action set. Fudenberg and Levine [11] proved that every such game, under specific assumptions, has a sequential equilibrium (in behavioral strategies). The class of games for which this holds is best described as extensive games with observable actions, simultaneous moves, perfect recall and a finite action set, plus some continuity requirements. As a consequence, the result in [11] does *not* directly apply to LCL games. Indeed, first, in LCL games the actions of far-away nodes are not observable. Second, the imperfect information in [11] is solely related to the fact that players play simultaneously, while, again, in LCL games, imperfect information also refers to the fact that each node is not aware of the states of far away nodes in the network. Before showing how to extend [11] to LCL games, we discuss next of the types of equilibria which should be sought in the context of LCL games.

3.2 Nash equilibria do not capture efficient distributed computation

We consider two important notions of Nash equilibria: the weakest form of equilibrium is the eponym Nash equilibrium while the strongest form of Nash equilibrium is the trembling-hand perfect equilibrium. The following result indicates that the basic notion of Nash equilibria is too weak as far as the design of *efficient* algorithms for LCL tasks is concerned.

Proposition 1 *For every $k \geq 1$, there is an LCL game $\Gamma^{(k)}$ for a task solvable in at most r rounds with probability at least $1 - \frac{1}{2^r}$ for every $r \geq 1$, and a Nash equilibria for $\Gamma^{(k)}$ which systematically requires at least k rounds to converge.*

Proof. We show that this negative result holds even for a task on the 2-node graph K_2 . Recall that, in a strategic game, a strategy s_i of player i is *strongly dominated* by another strategy t_i if, for

every strategy profile s'_{-i} of the other players, the respective payoffs satisfy $\pi_i(s_i, s'_{-i}) < \pi_i(t_i, s'_{-i})$. The strategy s_i is *weakly dominated* by t_i if $\pi_i(s_i, s'_{-i}) \leq \pi_i(t_i, s'_{-i})$ for every strategy profile s'_{-i} and $\pi_i(s_i, s''_{-i}) < \pi_i(t_i, s''_{-i})$ for at least one strategy profile s''_{-i} . Dominated strategies are undesirable, and it is usually assumed that rational players never play dominated strategies. Nevertheless, while Nash equilibria cannot contain strongly dominated strategies, they may contain weakly dominated strategies. The strategic game defined in Fig. 1(left) is an example of such an issue. We call this game the *constrained coloring* game. Each of the two players has choice between three frequencies, namely green (G), red (R), and blue (B). To maximize their payoffs, the players should better have one playing blue, and the other playing green.

	R	G	B
R	0, 0	0, 0	0, 0
G	0, 0	0, 0	1, 1
B	0, 0	1, 1	0, 0

	s^k	\hat{s}	s^0
s^k	δ^k	$\leq \delta^k$	δ^k
\hat{s}	$\leq \delta^k$		$\geq \delta^k$
s^0	δ^k	$\geq \delta^k$	1

Figure 1: *Left: The 2-player strategic game called constrained coloring; Right: Payoffs in the LCL game induced by constrained coloring*

The game defined in Fig. 1(left) has four Nash equilibria: (R, R) with payoffs $(0, 0)$, (G, B) with payoffs $(1, 1)$, (B, G) with payoffs $(1, 1)$, and $(\frac{G+B}{2}, \frac{G+B}{2})$ with payoffs $(\frac{1}{2}, \frac{1}{2})$. The Nash equilibrium (R, R) is not realistic since the strategy R is weakly dominated by both G and B . We show that such weakly dominated strategy are generally undesirable in LCL games since they can have dramatic consequences on the convergence time of the algorithm.

Consider the LCL game played by two players linked by an edge, inspired from the strategic game constrained coloring of Fig. 1(left). Every node has the choice between three colors: green, blue, and red. A radius-1 ball is good if and only if the two nodes have different colors. However, the cost of the good balls differ. A good ball including color red has low preference. All other good balls have higher preferences. Specifically, the two players have the same payoff function depending on their colors. At the first round, the payoffs are

- $(2 - \delta, 2 - \delta)$ when one player is green and the other is blue,
- (δ^k, δ^k) when one player is red and the other green or blue,

where $\delta \in (0, 1)$ is the discount factor of the game, and k is a positive integer parameter. In the case of a collision, another round of the game is played, and the payoffs are discounted by a factor δ .

The tree structure of the game is represented in Fig. 2, where the triangles are self-similar representation of the same tree, with payoff discounted by a multiplicative factor δ . The tree is thus infinite. The figure also describes the different payoffs after the first round.

As it can be read on Fig. 2, if the players choose their actions among (R, G) , (R, B) , (G, R) and (B, R) , then the game ends after the first round, and each player receives a payoff of δ^k . If the players choose their actions among (G, B) and (B, G) , then the game also ends, but each player receives a payoff of $2 - \delta$. Finally, if the players choose their actions among (R, R) , (G, G) and (B, B) , then game carries on, with payoffs discounted by a factor δ .

In this game, since each player is fully informed of the history of the play after any given round, a pure strategy of a player is simply an infinite sequence of actions. Let us denote by S the set of

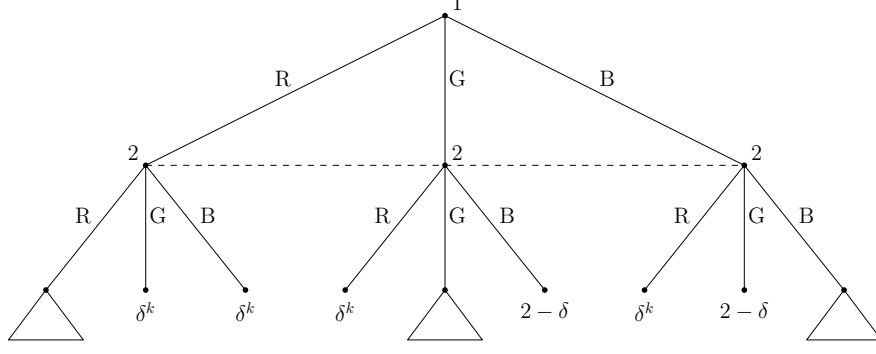


Figure 2: Tree structure of the LCL game inspired by constrained coloring

all pure strategies. A mixed strategy of a player is an infinite sequence of probability distributions over the set of actions $\{R, G, B\}$. Let us denote by \hat{S} the set of all mixed strategies. From Kuhn's theorem [17], it follows that \hat{S} is actually equal to the set of all probability distributions over S . We will pay particular attention to the strategies

$$s^l = (\underbrace{R, \dots, R}_l, \frac{G+B}{2}, \frac{G+B}{2}, \dots),$$

which consist in playing the action R for the first l rounds, and then playing indefinitely the mixed strategy $\frac{G+B}{2}$. All sequences are indexed from 0. For every sequence s , and for every positive integer i , the sequence $s_{|i}$ denotes the sequence resulting from s after removing the i first actions, and reindexing the actions $(i, i+1, \dots)$ in $s_{|i}$ as $(0, 1, \dots)$. Let us now compute the equilibria in the LCL game induced by constrained coloring. We denote by $\Pi : S^2 \mapsto \mathbb{R}$ the expected payoff function of the game (note that the players always receive the same payoff).

Fact 1 *Let $s \in S$ be a pure strategy. If R is never played in s , then $\Pi(s^0, s) = \Pi(s, s^0) = 1$. Otherwise, let t be the first round in which the action R is played in s . We have*

$$\Pi(s^0, s) = \Pi(s, s^0) = 1 - \left(\frac{\delta}{2}\right)^t (1 - \delta^k).$$

Indeed, if the first action in s is R , then the game ends in the first round, and $\Pi(s, s^0) = \delta^k$. So, let us assume that the first action in s is G or B . Then, with probability $\frac{1}{2}$, the game ends, and each player gets payoff $2 - \delta$. Similarly, with probability $\frac{1}{2}$, another round is played. Since $s_{|1}^0 = s^0$, we get the following recurrence relation:

$$\Pi(s, s^0) = \frac{1}{2} (2 - \delta) + \frac{1}{2} \delta \Pi(s_{|1}, s^0).$$

This recurrence yields

$$\Pi(s, s^0) = \frac{2 - \delta}{2} \sum_{i=0}^{t-1} \left(\frac{\delta}{2}\right)^i + \left(\frac{\delta}{2}\right)^t \delta^k = \frac{2 - \delta}{2} \left(\frac{1 - (\frac{\delta}{2})^t}{1 - \frac{\delta}{2}}\right) + \left(\frac{\delta}{2}\right)^t \delta^k = 1 - (1 - \delta^k) \left(\frac{\delta}{2}\right)^t$$

as desired.

Fact 2 Let $s \in S$ be a pure strategy. Let t' be the first round at which action G or B is played in s , and let t'' be the first round at which action R is played in $s|_k$.

$$\Pi(s^k, s) = \Pi(s, s^k) = \begin{cases} \delta^{t'+k} & \text{if } t' < k \\ \delta^{2k} & \text{if } t' > k \\ \delta^k \left(1 - \left(\frac{\delta}{2}\right)^{t''} (1 - \delta^k)\right) & \text{if } t' = k \end{cases}$$

Indeed, if the first action in s is G or B , then the game ends in the first round and $\Pi(s, s^k) = \delta^k$. Let us assume that the first action in s is R , then the next round is played, and $\Pi(s, s^k) = \delta(s|_1, s|_1)$. Fact 2 then follows from Fact 1.

Note that it was sufficient to establish the above facts for pure strategies since, for any two mixed strategies \hat{s}_1 and \hat{s}_2 , there exist two pure strategies s and s' in the support of \hat{s}_1 such that $\Pi(s, \hat{s}_2) \leq \Pi(\hat{s}_1, \hat{s}_2) \leq \Pi(s', \hat{s}_2)$. The following fact follows from Fact 2, because s^k is a best response to itself as it yields the highest payoff δ^k .

Fact 3 The strategy profile (s^k, s^k) is a Nash equilibrium.

Finally, the following completes our analysis of the LCL game induced by constrained coloring.

Fact 4 The strategy s^k is (weakly) dominated by the strategy s^0 .

To see why the fact holds, let $s \in S$ be a pure strategy and t be the first round in which the action R is played in s . Since $\delta < 1$, we get $\delta^k \leq 1 - (1 - \delta^k) \left(\frac{\delta}{2}\right)^t$, from which it follows from Fact 1 that $\Pi(s^0, s) \geq \delta^k$. Furthermore, from Fact 2, we get that $\Pi(s^k, s) \leq \delta^k$. By extension to mixed strategies, we have that, for every mixed strategy $\hat{s} \in \hat{S}$, we have $\Pi(s^0, \hat{s}) \geq \Pi(s^k, \hat{s})$. It remains to show that there exists a mixed strategy $\hat{s} \in \hat{S}$ such that $\Pi(s^0, \hat{s}) > \Pi(s^k, \hat{s})$. This is achieved for example by $\hat{s} = s^0$. Indeed by Fact 1, we have $\Pi(s^0, s^0) = 1$ and $\Pi(s^k, s^0) = \delta^k$ which completes the proof of Fact 4.

The above claims are summarized in Fig. 1(right), where \hat{s} represents any mixed strategy not in $\{s^0, s^k\}$. The strategy profile (s^k, s^k) is a Nash equilibrium. For this strategy profile, the LCL game induced by constrained coloring requires at least k rounds to converge because both players systematically play action R for the first k rounds. On the other hand, The strategy profile (s^0, s^0) allows convergence in constant expected number of rounds, and converges in at most r rounds with probability at least $1 - \frac{1}{2^r}$. \square

3.3 Trembling-hand perfect equilibria do capture efficient distributed computation

We have seen in Proposition 1 that focussing solely on Nash equilibria in a broad sense is not sufficient. Indeed, such equilibria can include weakly dominated strategies, and this may have a dramatic impact on the convergence time of the algorithm. Such problem can be eliminated by considering the stronger notion of trembling-hand perfect equilibrium (a subset of Nash equilibria). Indeed, recall that a trembling-hand perfect equilibrium of a game Γ is a Nash equilibrium which is the limit of Nash equilibria of perturbed games Γ_ϵ , $\epsilon > 0$, where Γ_ϵ is essentially the game Γ but in which the players can set their strategies only up to some error ϵ . It is known that trembling-hand

perfect equilibria avoid pathological scenarios in general extensive games [11, 30], mostly because these equilibria never include weakly dominated strategies, as a weakly dominated strategy can never be a best response to a completely mixed strategy profile. In other words, in the perturbed game Γ_ϵ , all players must give a probability at least ϵ to every action. For instance, in the perturbed LCL game for MIS, no nodes can systematically propose not to be in the MIS, and must propose to enter the MIS with probability at least ϵ at each round. Similarly, in the perturbed LCL game for $(\Delta + 1)$ -coloring, every color must have a probability at least ϵ to be proposed. As a third example, one can show that, for the LCL games $\Gamma^{(k)}$, $k \geq 1$, whose existences are established in the proof of Proposition 1, the trembling-hand perfect equilibria yield algorithms converging quickly. More specifically, we have the following:

Proposition 2 *For all the LCL games $\Gamma^{(k)}$, $k \geq 1$, on the 2-node graph K_2 whose existences are established in Proposition 1, the trembling-hand perfect equilibria yield algorithms converging in at most r rounds, with probability at least $1 - \frac{1}{2^r}$, for any $r \geq 1$.*

Proof. Fig. 3 illustrates the notion of trembling-hand perfect equilibria for the LCL game constrained coloring defined in the proof of Proposition 1. In Fig. 3, any point in the triangle BRG represents a distribution D of probability in the set $\{B, R, G\}$ of colors. In Γ_ϵ , these distributions are bounded to be at least ϵ away from the boundaries, hence they are inside the triangle with dotted line borders. In Γ_ϵ , all equilibria correspond to strategy profiles where the strategy of every player stands on the bottom line marked in bold in this latter triangle. Hence, the Nash equilibrium (s^k, s^k) cannot be the limit, when ϵ goes to 0, of equilibria in Γ_ϵ . Indeed, the corner R cannot be the limit of points in the bold line, as this line actually converges to the line GB. Therefore, the undesirable Nash equilibrium (s^k, s^k) is not a trembling-hand perfect equilibrium of the LCL game for constrained coloring.

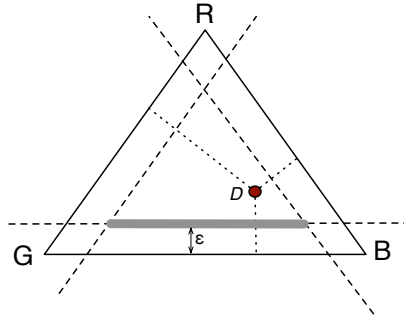


Figure 3: Illustration of trembling hand perfect equilibrium

On the other hand, we know from Fact 4 that s^k is weakly dominated by s^0 . In particular, the strategy profile (s^0, s^0) is a trembling-hand perfect equilibrium for the game stated in Proposition 1, and this strategy profile allows the game to converge in a constant expected number of rounds. \square

In general, in a perturbed LCL game, no player can block the game for an arbitrarily long time by strictly refusing to select a solution that does not please it. Now, can we always guarantee the existence of trembling hand perfect equilibria in LCL games? In the next sections, we show that the answer to this question is positive.

4 Existence of trembling-hand perfect equilibria

In this section, for the purpose of establishing the existence of trembling-hand perfect equilibria for LCL-games, we extend the result of [11] by allowing actions of other players be not observable, which also requires to generalize the notion of rounds as well as the notion of imperfect information, such as used in [11].

Lemma 1 *Every infinite, continuous, measurable, well-rounded, symmetric extensive game with perfect recall and finite action set has a symmetric trembling-hand perfect equilibrium.*

The rest of the section is dedicated to the proof of Lemma 1. We start by formally defining the model.

4.1 The Model

We focus our attention on *extensive games* with imperfect information. Our objective is to include *infinite horizon* in the analysis of such games. Recall that an extensive game Γ is a tuple (N, A, X, P, U, p, π) , where:

- $N = \{1, \dots, n\}$ is the set representing the *players* of the game. An additional player, denoted by c , and called *chance*, is modeling all random effects that might occur in the course of the game.
- A is the (finite) *action set*, i.e., a finite set representing the actions that can be made by each player when she has to play.
- X is the *game tree*, that is, a subset of $A^* \cup A^\omega$ where A^* (resp., A^ω) denotes the set of finite (resp., infinite) strings with elements in A , satisfying the following properties:
 - the empty sequence $\emptyset \in X$;
 - X is stable by prefix;
 - if $(a_i)_{i=1, \dots, k} \in X$ for every $k \geq 1$, then $(a_i)_{i \geq 1} \in X$.

The set X is partially ordered by the prefix relation, denoted by \preceq , where $x \preceq y$ means that x is a prefix of y , and $x \prec y$ means that x is a prefix of y distinct from y . The elements of X are called *histories*. A history x is *terminal* if it is a prefix of no other histories in X . In particular, every infinite history in X is terminal. The set of terminal histories is denoted by Z . If the longest history is finite then the game has *finite horizon*, otherwise it has *infinite horizon*. For every non-terminal history x , we denote by

$$A(x) = \{a \in A : (x, a) \in X\}$$

the set of available actions after the history x .

- P is the *player partition*, i.e., a function $P : X \setminus Z \rightarrow N \cup \{c\}$ that assigns a player to each non-terminal history. $P(x)$ is the player who has to play after the history x . The sets $P_i = \{x \in X \setminus Z : P(x) = i\}$, $i \in N \cup \{c\}$, called *player sets*, form a partition of $X \setminus Z$.

- U is the *information partition*, that is, a refinement of the player partition whose elements are called *information sets* such that for every $u \in U$, and for every two histories x, y in this *information set* u , we have $A(x) = A(y)$, i.e., the sets of available actions after x and y are identical. We can therefore define $A(u)$ as the set of actions available after the information set u . Formally,

$$A(u) = \{a \in A : (x, a) \in X \text{ for every } x \in u\}.$$

For every history x , the information set containing x is denoted by $u(x)$. We also define $P(u)$ as the player who has to play after the information set u has been reached, and for every player i , the set $U_i = \{u \in U : P(u) = i\}$. $\{U_i, i \in N \cup \{c\}\}$ forms a partition of U . Information sets regroup histories that are indistinguishable to players. Since the player chance is not expected to behave rationally, we will simply put $U_c = \{\{x\}, x \in P_c\}$.

- p is a function that assigns to every history x in P_c (the player set of the chance) a probability distribution over the set $A(x)$ of available actions after the history x . This *chance function* p is supposed to be common knowledge among the players.
- π is the *payoff function*, that is, $\pi : Z \rightarrow \mathbb{R}^n$ assigns the payoff (a real value) to every player in N for every terminal history of the game. We assume that every payoff is in $[-M, M]$ for some $M \geq 0$.

We introduce the concept of *rounds* in extensive games as follows.

Definition 1 *The round function r of an extensive game assigns a positive integer to every non terminal history x , defined by $r(x) = |\text{Rec}(x)|$ where $\text{Rec}(x) = \{x' \in X \mid x' \prec x \text{ and } P(x') = P(x)\}$. We call $r(x)$ the round of x . The round of a finite terminal history is the round of its predecessor, and the round of an infinite history z is $r(z) = \infty$. An extensive game Γ for which the round function is non decreasing with respect to the prefix relation, i.e $y \preceq x \Rightarrow r(y) \leq r(x)$, is said to be well-rounded.*

Note that not every game is well-rounded, because two histories x and y such that $x \preceq y$ do not necessarily verify $P(x) = P(y)$. In a well-rounded game, since r is non decreasing, we have that for any non terminal history x , every player has played at most $r(x) + 1$ times before x . Moreover, every player which has played less than $r(x)$ times before x will never play again after x .

Let $u \in U_i$ and $u' \in U_i$ be two (non necessarily distinct) information sets of the same player i , for which there exist $x \in u$, $x' \in u'$, and $a \in A(u')$, such that $(x', a) \preceq x$. Recall that an extensive game is said to have *perfect recall* if, for every such i , u , u' and a , we have:

$$\forall y \in u, \exists y' \in u' \mid (y', a) \preceq y.$$

The following lemma will allow us to safely talk about the round of an information set.

Lemma 2 *Let Γ be an extensive game with perfect recall, and let $x \in X$ and $x' \in X$ be two non terminal histories in the same information set $u \in U$. Then x and x' have the same round.*

Proof. We first observe the following. Let Γ be an extensive game with perfect recall, and let $y \in X$ be a finite history. Let $y' \in X$ and $y'' \in X$ for which there is $u \in U$ such that $y' \in \text{Rec}(y) \cap u$ and $y'' \in \text{Rec}(y) \cap u$. Then $y' = y''$. Indeed, since both y' and y'' are in $\text{Rec}(y)$, we have that both

are prefixes of y , and thus one of the two is a prefix of the other. Assume, w.l.o.g., that $y'' \prec y' \prec y$ (as, if $y' = y''$ then we are done). Let a be the action such that $(y'', a) \preceq y'$. Since the game has perfect recall, there must exist a history $y''' \in u$ such that $(y''', a) \preceq y''$. Thus $y''' \prec y'' \prec y' \prec y$. We can repeat the same reasoning for y''' and y'' as we did for y'' and y' , so on and so forth. In this way, we construct an infinite strictly decreasing sequence of histories, which contradicts the fact that y is finite.

If both $Rec(x)$ and $Rec(x')$ are empty, then x and x' have the same round. Assume, w.l.o.g., that $Rec(x) \neq \emptyset$, and let $y \in Rec(x)$. Let a be the action such that $(y, a) \preceq x$. Since the game has perfect recall, there exists $y' \in u(x')$ such that $(y', a) \preceq x'$. Therefore $y' \prec x'$ and $P(y') = P(y) = P(x) = P(x')$. It follows that $y' \in Rec(x')$. Thus, for any $y \in Rec(x)$, we have found an $y' \in Rec(x')$. This mapping from $Rec(x)$ to $Rec(x')$ is one to one. Indeed, let y_1 and y_2 in $Rec(x)$, and let y'_1 and y'_2 in $Rec(x')$ be the corresponding histories. If $y'_1 = y'_2$, then, since $u(y_1) = u(y'_1)$ and $u(y_2) = u(y'_2) = u(y'_1)$, we get that $y_1 \in Rec(x) \cap u(y'_1)$, $y_2 \in Rec(x) \cap u(y'_1)$. It follows from the above observation that $y_1 = y_2$. Thus the mapping is one-to-one, and hence $r(x) \leq r(x')$. It follows that we also have $Rec(x') \neq \emptyset$. Therefore, we can apply the same reasoning by switching the roles of x and x' , which yields $r(x') \leq r(x)$. Thus $r(x) = r(x')$. \square

4.2 Strategies, Outcomes and Expected Payoff

In this section, we first recall several basic concepts about extensive games with perfect recall. Without loss of generality, we restrict our attention to *behavioral strategies* since such strategies are outcome-equivalent to mixed strategies thanks to Kuhn's theorem [18]. The main objective of this section is to define the *expected payoff function*, which is novel for it is adapted to the infinite games.

Recall that the *local strategy* $b_{i,u}$ of a player i given the information sets u is a probability distribution over the set $A(u)$ of actions available given this information set. The set of local strategies of player i for u is denoted by $B_{i,u}$. The *behavioral strategy* b_i of a player i is a function which assigns a local strategy $b_{i,u}$ to every information set u of this player. The set of all behavioral strategies of player i is denoted by B_i . A *strategy profile* is a n -tuple of behavioral strategies, one for each player. The set of all strategy profiles is $B = \times_{i \in N} B_i$. For each player i , we denote by B_{-i} the set $\times_{j \neq i} B_j$. Since $B = B_i \times B_{-i} = \times_{i \in N} B_i$, a strategy profile b can be identified in these different ways as $b = (b_i, b_{-i}) = (b_1, b_2, \dots, b_n)$. When every player plays according to a strategy profile b , the outcome of the game is entirely determined, in the sense that every history x has a probability $\rho_b(x)$ of being reached. For any strategy profile b and any history $x = (a_i)_{i=1, \dots, k}$ where $k \in \mathbb{N} \cup \infty$, the *realization probability* of x is defined by $\rho_b(\emptyset) = 1$, and

$$\rho_b(x) = \prod_{i=0}^{k-1} b_{P(x_i), u(x_i)}(a_{i+1})$$

where $x_0 = \emptyset$, and, for every positive $i \leq k$, $x_i = (a_1, a_2, \dots, a_i)$. When the player is the chance, we simply identify its strategy with the chance function p : $P(x_i) = c \Rightarrow b_{P(x_i), u(x_i)}(a_{i+1}) = p(x_i, a_{i+1})$. The function $\rho_b : X \rightarrow [0, 1]$ is called the *outcome* of the game under the strategy profile b . An outcome $\rho : X \rightarrow [0, 1]$ is *feasible* if and only if there exists a strategy profile b such that $\rho = \rho_b$. The set of feasible outcomes of Γ is denoted by O .

We are now ready to define the *expected payoff*. Note that in a game with infinite horizon, there can be uncountably many terminal histories. Therefore the definition of the probability measure

on Z requires some care. For any finite history x , let Z_x be the set $\{z \in Z \mid x \preceq z\}$. Note that Z_x might be uncountable. Let Σ be the σ -algebra on Z generated by all sets of the form Z_x for some finite history x . For each strategy profile b , the measure μ_b on Σ is defined by: for every set Z_x , $\mu_b(Z_x) = \rho_b(x)$.

This definition ensures that μ_b is a probability measure because $\mu_b(Z) = \mu_b(Z_\emptyset) = \rho_b(\emptyset) = 1$.

Definition 2 Let π be a payoff function that is measurable on Σ . The expected payoff function Π assigns a real value $\Pi(b)$ to every strategy profile $b \in B$, defined by

$$\Pi(b) = \int_{\Sigma} \pi \, d\mu_b .$$

Note that each component of the expected payoff function is bounded by M , where M is the upper bound on every payoff. A game Γ whose payoff function π is measurable on Σ is said to be a *measurable game*. In the following, we always assume that the considered games are measurable.

4.3 ϵ -Equilibria and Subgame Perfection

We now show how to adapt the standard notion of ϵ -equilibria (cf., e.g., Radner [28]) to infinite games (again, Nash equilibrium is a special case of ϵ -equilibrium, with $\epsilon = 0$). Recall that a strategy profile b is a ϵ -equilibrium if and only if, for every player i , and every behavior strategy $b'_i \in B_i$ of this player, we have $\Pi_i(b'_i, b_{-i}) - \Pi_i(b) \leq \epsilon$. Similarly, we recall the notions of *subgames* and *subgame perfect equilibria* (see, e.g., [29]). A subtree X' of X is said to be *regular* if no information sets contain both a history in X' and a history not in X' . To each regular subtree X' is associated a game $\Gamma' = (N, A, X', P', U', p', \pi')$, where P' , U' , p' and π' are the restrictions of P , U , p and π to X' , called a *subgame*. The notions of outcomes and expected payoff functions for subgames follow naturally. Specifically,

Definition 3 A strategy profile b is a subgame perfect ϵ -equilibrium of an infinite game Γ if and only if, for every subgame Γ' , the restriction of b to Γ' is an ϵ -equilibrium.

Note that a subgame perfect ϵ -equilibrium of Γ is an ϵ -equilibrium since Γ is a subgame of itself.

4.4 Metrics

In this section, we now define specific metrics on the set O of feasible outcomes, and on the set of behavior strategy profiles. These definitions are inspired from [11], with adaptations to fit our infinite setting.

Definition 4 Let $\rho^1, \rho^2 \in O$ be two feasible outcomes of the same extensive game Γ . We define the following metric d on O :

$$d(\rho^1, \rho^2) = \sup_{\substack{x \in X \\ x \text{ finite}}} 2^{-r(x)} \cdot |\rho^1(x) - \rho^2(x)|$$

where $r(x)$ is the round of the finite history x .

Lemma 3 The function $d : O \times O \rightarrow \mathbb{R}$ of Definition 4 is indeed a metric.

Proof. We first show that d satisfies the triangle inequality. Let ρ^1, ρ^2 and ρ^3 be three feasible outcomes of Γ . For any finite history x we have the following:

$$\begin{aligned}
& |\rho^1(x) - \rho^3(x)| \leq |\rho^1(x) - \rho^2(x)| + |\rho^2(x) - \rho^3(x)| \\
\Rightarrow & 2^{-r(x)} |\rho^1(x) - \rho^3(x)| \leq 2^{-r(x)} |\rho^1(x) - \rho^2(x)| + 2^{-r(x)} |\rho^2(x) - \rho^3(x)| \\
\Rightarrow & \sup_{\substack{x \in X \\ x \text{ finite}}} 2^{-r(x)} |\rho^1(x) - \rho^3(x)| \leq \sup_{\substack{x \in X \\ x \text{ finite}}} \left(2^{-r(x)} |\rho^1(x) - \rho^2(x)| + 2^{-r(x)} |\rho^2(x) - \rho^3(x)| \right) \\
\Rightarrow & \sup_{\substack{x \in X \\ x \text{ finite}}} 2^{-r(x)} |\rho^1(x) - \rho^3(x)| \leq \sup_{\substack{x \in X \\ x \text{ finite}}} 2^{-r(x)} |\rho^1(x) - \rho^2(x)| + \sup_{\substack{x \in X \\ x \text{ finite}}} 2^{-r(x)} |\rho^2(x) - \rho^3(x)| \\
\Rightarrow & d(\rho^1, \rho^3) \leq d(\rho^1, \rho^2) + d(\rho^2, \rho^3)
\end{aligned}$$

Next, we prove that d separates different outcomes. Let ρ^1 and ρ^2 be two feasible outcomes such that $d(\rho^1, \rho^2) = 0$. By definition, this implies that, for every finite history x , we have $\rho^1(x) = \rho^2(x)$. Let b^1 and b^2 be two strategy profiles such that $\rho^1 = \rho_{b^1}$ and $\rho^2 = \rho_{b^2}$. Let $x = (a_k)_{k \geq 1}$ be an infinite history, and, for $k \geq 1$, let $x_k = (a_1, a_2, \dots, a_k)$ be the corresponding increasing sequence of its prefixes. By definition, $\rho_{b^1}(x)$ is the limit for $k \rightarrow \infty$ of the sequence $\rho_{b^1}(x_k)$, and $\rho_{b^2}(x_k) \rightarrow \rho_{b^2}(x)$ for $k \rightarrow \infty$ as well. Since the two sequences are equal they have the same limit, and therefore $\rho^1(x) = \rho^2(x)$. Since this is true for any infinite history x , it follows that $\rho^1 = \rho^2$. \square

We can then use d to define a metric on behavioral strategy profiles as follows. Let $b^1, b^2 \in B$ be two behavioral strategy profiles of the same game Γ . We define the metric d on B by:

$$d(b^1, b^2) = \max \left\{ d(\rho_{b^1}, \rho_{b^2}), \sup_{\substack{i \in N \\ b_i \in B_i}} d(\rho_{(b_i, b_{-i}^1)}, \rho_{(b_i, b_{-i}^2)}) \right\}$$

Finally, we define the continuity of the expected payoff function using the sup norm over \mathbb{R}^n . Specifically, the expected payoff function Π is *continuous* if, for every sequence of strategy profiles $(b^k)_{k \geq 1}$ and every strategy profile b , we have

$$d(b^k, b) \xrightarrow{k \rightarrow \infty} 0 \implies \sup_{i \in N} |\Pi_i(b^k) - \Pi_i(b)| \xrightarrow{k \rightarrow \infty} 0$$

We then say that an extensive game Γ is continuous if its expected payoff function is continuous.

4.5 Perturbed Games and Perfect Equilibria

In this section, we extend the classical notion of trembling-hand perfect ϵ -equilibria to infinite games, generalizing [30]. We start by revisiting the notion of a perturbed game in the context of infinite games. A *perturbed game* $\hat{\Gamma}$ is a pair (Γ, η) , where Γ is an extensive game and η is a function which assigns a positive probability $\eta_u(a)$ to *every* action a available after the information set u such that

$$\forall u \in U, \quad \sum_{a \in A(u)} \eta_u(a) < 1. \quad (1)$$

The perturbed game $\hat{\Gamma}$ is to be interpreted as a restriction of the game Γ in the following sense. The probability $\eta_u(a)$ is the *minimal probability* that player $P(u)$ can assign to the action a in his local strategy at information set u . Therefore we define the set of strategies of the perturbed game

as follows. In the perturbed game $\hat{\Gamma}$, a *local strategy* $\hat{b}_{i,u}$ of a player i at one of its information sets u is a probability distribution over the set $A(u)$ of actions available at this information set satisfying:

$$\forall (i, u, a) \in N \times U_i \times A(u), \quad \hat{b}_{i,u}(a) \geq \eta_u(a).$$

The set of local strategies of player i at the information set u in the perturbed game $\hat{\Gamma}$ is denoted by $\hat{B}_{i,u}$. Note that $\hat{B}_{i,u}$ is not reduced to a single strategy thanks to Eq. (1).

From the above, we can define the set \hat{B}_i of behavioral strategies of player i , the set \hat{B} of strategy profiles, and the notion of ϵ -equilibria of a perturbed game in the same way as for regular extensive games.

Definition 5 *A strategy profile b^* is a trembling hand perfect ϵ -equilibrium of Γ if*

1. *there exists a sequence $(\hat{\Gamma}^k) = (\Gamma, \eta^k)$, $k \geq 1$, of perturbations of Γ such that $\eta^k \rightarrow 0$ when $k \rightarrow \infty$ (where $\eta^k \rightarrow 0$ means that, for every player i , every information set $u \in U_i$, and every action $a \in A(u)$, $\eta_u^k(a) \rightarrow 0$), and*
2. *there exists a sequence of ϵ -equilibria \hat{b}^k of these perturbed games such that $\hat{b}^k \rightarrow b^*$ when $k \rightarrow \infty$ (where $\hat{b}^k \rightarrow b^*$ in the sense of the metric on behavioral strategy profiles).*

Such a sequence $(\hat{\Gamma}^k)_{k \geq 1}$ of perturbed games is called a test sequence for Γ .

The next lemma shows that the notion of trembling hand perfect ϵ -equilibrium is a refinement of the notion of subgame perfect ϵ -equilibrium.

Lemma 4 *Every trembling hand perfect ϵ -equilibrium of a continuous game is a subgame perfect ϵ -equilibrium.*

Proof. We first show that every trembling hand perfect ϵ -equilibrium of a continuous game is an ϵ -equilibrium. Let b^* , Γ , $\hat{\Gamma}^k$ and \hat{b}^k be as in Definition 5. Since \hat{b}^k are ϵ -equilibria of the games $\hat{\Gamma}^k$, the following holds:

$$\forall i \in N, \forall k, \forall b_i \in \hat{B}_i^k, \quad \Pi_i(b_i, \hat{b}_{-i}^k) - \Pi_i(\hat{b}^k) \leq \epsilon$$

Let B_i^m be the intersection of all \hat{B}_i^k with $k \geq m$. Then we have:

$$\forall i \in N, \forall k \geq m, \forall b_i \in B_i^m, \quad \Pi_i(b_i, \hat{b}_{-i}^k) - \Pi_i(\hat{b}^k) \leq \epsilon$$

Since Π is assumed to be continuous, these inequalities remain valid if on both sides, we take the limits for $k \rightarrow \infty$:

$$\forall i \in N, \forall b_i \in B_i^m, \quad \Pi_i(b_i, b_{-i}^*) - \Pi_i(b^*) \leq \epsilon$$

This holds for every m . The closure of the union of all B_i^m is B_i . Since Π is continuous, this yields:

$$\forall i \in N, \forall b_i \in B_i, \quad \Pi_i(b_i, b_{-i}^*) - \Pi_i(b^*) \leq \epsilon$$

as desired.

Moreover, every ϵ -equilibrium of a perturbed game is subgame perfect. Indeed, this follows from the fact that in a perturbed game, every subgame is reached with a strictly positive probability.

We now have all the ingredients to establish the lemma. Let b^* be a trembling hand perfect ϵ -equilibrium of the game Γ and \hat{b}_k a sequence of equilibria of the test sequence $\hat{\Gamma}$ such that $\hat{b}^k \rightarrow b^*$.

Every subgame Γ' of Γ induces a sequence of subgames $\hat{\Gamma}'^k$ of the test sequence. Since every ϵ -equilibrium of a perturbed game is subgame perfect, it follows that the sequence \hat{b}^k induces a sequence of equilibria \hat{b}'^k of these perturbed subgames, which converges to b'^* , the strategy profile induced by b^* on Γ' . Therefore b'^* is a trembling hand perfect ϵ -equilibrium of Γ' , which proves that b^* is subgame perfect since every trembling hand perfect ϵ -equilibrium of a continuous game is an ϵ -equilibrium. \square

4.6 Truncated Games and Induced Equilibria

The goal of this section is to introduce the concept of *truncation* of an infinite game, and to draw links between equilibria of the truncated (finite) game and the equilibria of the corresponding infinite game. The purpose of truncation is to deal with finite game that are easier to handle than infinite ones. The material in this section is a generalization of [11] to extensive games (the original paper [11] only applies to repeated games).

For every well-rounded extensive game $\Gamma = (N, A, X, P, U, p, \pi)$ with perfect recall, and for every positive integer t , we define the *truncated game* $\Gamma_t = (N, A, X_t, \tilde{P}, \tilde{U}, \tilde{p}, \tilde{\Pi})$ as follows.

- X_t is the set of all histories $x \in X$ of round at most t . The set of terminal histories \tilde{Z} of Γ_t is defined as the set of all histories with a predecessor in X_t and no successor in X_t .
- \tilde{P} , \tilde{U} and \tilde{p} are the restrictions of P , U , and p to X_t . Note that these restrictions do not break information sets thanks to Lemma 2.
- Strategies and strategy profiles of Γ are mapped to strategies and strategy profiles of Γ_t as follows.
 - Every strategy profile b of Γ is mapped to a strategy profile \tilde{b} of Γ_t obtained by the restriction of b to the information sets in X_t . The set of strategy profiles of Γ_t is denoted by \tilde{B} . The strategy profile \tilde{b} is said to be *induced* by b .
 - Conversely, every strategy profile \tilde{b} of Γ_t is mapped to the strategy profile b of Γ defined by
 - * $b_{i,u} = \tilde{b}_{i,u}$ for every information set $u \in \tilde{U}$ and every player i ,
 - * $b_{i,u}(a) = \frac{1}{|A(u)|}$ for every information set $u \notin \tilde{U}$, every player i and every action a (i.e., $b_{i,u}$ is a uniform distribution).

The strategy profile b is said to be induced by \tilde{b} .

Note that the first induction mapping is not necessarily one-to-one, while the second is not necessarily onto.

- In order to define the new payoff function $\tilde{\pi}$ over the set \tilde{Z} , we rather define it implicitly, via the expected payoff function $\tilde{\Pi}$ which is set such that, for any strategy profile $\tilde{b} \in \tilde{B}$ that induces $b \in B$, we have $\tilde{\Pi}(\tilde{b}) = \Pi(b)$.

Lemma 5 *In a continuous well-rounded game with perfect recall Γ , a strategy profile b^* is an ϵ -equilibrium if and only if there exists sequences ϵ^k , t^k and b^k such that for every k , b^k is an ϵ^k -equilibrium in Γ_{t^k} and as k goes to infinity, $\epsilon^k \rightarrow \epsilon$, $t^k \rightarrow \infty$ and $b^{t^k} \rightarrow b^*$, where b^{t^k} is induced by b^k on Γ .*

Proof. Given a game Γ , for every $t \geq 1$, let

$$w^t = \max_{i \in N} \sup_{b^1, b^2 \in B : \tilde{b}^1 = \tilde{b}^2} |\Pi_i(b^1) - \Pi_i(b^2)|$$

where $\tilde{b}^1 = \tilde{b}^2$ stands for the fact that b^1 and b^2 induce the same strategy profiles over the truncated game Γ_t . Note that $w^t \leq 2M$ for every $t \geq 1$.

Claim 1 *A continuous game Γ satisfies $w^t \xrightarrow[t \rightarrow \infty]{} 0$.*

To establish the claim, let b^1 and b^2 be two strategy profiles of Γ which induce the same strategy profiles over Γ_t . This implies that for any history x of round at most t and any deviation b_i of some player i , the relations $\rho_{b^1}(x) = \rho_{b^2}(x)$ and $\rho_{(b_i, b_{-i}^1)}(x) = \rho_{(b_i, b_{-i}^2)}(x)$ holds. Therefore by the definition of the metric d on feasible outcomes (cf. Definition 4) and its extension to behavioral strategy profiles, we have $d(b^1, b^2) \leq 2^{-(t+1)}$. Therefore:

$$w^t \leq \sup_{\substack{i \in n \\ b^1, b^2 \in B \\ d(b^1, b^2) \leq 2^{-(t+1)}}} |\Pi_i(b^1) - \Pi_i(b^2)|$$

And since Γ is continuous,

$$\sup_{\substack{i \in n \\ b^1, b^2 \in B \\ d(b^1, b^2) \leq 2^{-(t+1)}}} |\Pi_i(b^1) - \Pi_i(b^2)| \xrightarrow[t \rightarrow \infty]{} 0$$

This completes the proof of Claim 1.

Claim 2 *Let Γ be a well-rounded game with perfect recall and Γ_t its truncation. We have that:*

1. *Any ϵ -equilibrium \tilde{b}^* in Γ_t induces a $(\epsilon + w^t)$ -equilibrium in Γ .*
2. *Any ϵ -equilibrium b^* in Γ induces a $(\epsilon + 2w^t)$ -equilibrium in Γ_t .*

To establish the first item of the claim, let \tilde{b}^* be an ϵ -equilibrium in Γ_t which induces b^* in Γ . Let i be a player, $b'_i \in B_i$ a deviation of this player in Γ , \tilde{b}_i the deviation it induces in Γ_t and b_i the deviation induced by \tilde{b}_i in Γ . Since \tilde{b}^* is an ϵ -equilibrium in Γ_t , we obtain that

$$\tilde{\Pi}_i(\tilde{b}_i, \tilde{b}_{-i}^*) - \tilde{\Pi}_i(\tilde{b}^*) \leq \epsilon$$

By definition of $\tilde{\Pi}$, the following holds:

$$\begin{aligned} \tilde{\Pi}_i(\tilde{b}_i, \tilde{b}_{-i}^*) &= \Pi_i(b_i, b_{-i}^*) \\ \tilde{\Pi}_i(\tilde{b}^*) &= \Pi_i(b^*) \end{aligned}$$

Furthermore, by definition of w^t , we have

$$\Pi_i(b'_i, b_{-i}^*) - \Pi_i(b_i, b_{-i}^*) \leq w^t$$

We eventually arrive at

$$\Pi_i(b'_i, b_{-i}^*) - \Pi_i(b^*) \leq \epsilon + w^t$$

Which proves that b^* is a $(\epsilon + w^t)$ -equilibrium, since i is any player and b'_i is any deviation in B_i .

To establish the second item of Claim 2, let b^* be an ϵ -equilibrium in Γ , \tilde{b}^* the strategy profile it induces on Γ_t , and b'^* the strategy profile induced by \tilde{b}^* on Γ . Let i be a player, $\tilde{b}_i \in \tilde{B}_i$ a deviation in Γ_t and b_i the deviation it induces in Γ . Because b^* is an ϵ -equilibrium, we have:

$$\Pi_i(b_i, b_{-i}^*) - \Pi_i(b^*) \leq \epsilon$$

By definition of $\tilde{\Pi}$, we have:

$$\begin{aligned}\tilde{\Pi}_i(\tilde{b}^*) &= \Pi_i(b'^*) \\ \tilde{\Pi}_i(\tilde{b}_i, \tilde{b}_{-i}^*) &= \Pi_i(b_i, b_{-i}^*)\end{aligned}$$

And by definition of w^t :

$$\begin{aligned}\Pi_i(b'^*) - \Pi_i(b^*) &\leq w^t \\ \Pi_i(b_i, b_{-i}^*) - \Pi_i(b_i, b_{-i}^*) &\leq w^t\end{aligned}$$

The above relations put together eventually gives us:

$$\tilde{\Pi}_i(\tilde{b}_i, \tilde{b}_{-i}^*) - \tilde{\Pi}_i(\tilde{b}^*) \leq \epsilon + 2w^t$$

Wich proves that \tilde{b}^* is a $(\epsilon + 2w^t)$ -equilibrium in Γ_t . This completes the proof of Claim 2.

Note that Claim 2 applies to perturbed games as well, as long as induced strategy profiles verify the constraints of the minimal probabilities.

Claim 3 *Let b^k be a sequence of ϵ -equilibria of a continuous well-rounded game with perfect recall Γ such that $b^k \rightarrow b^*$. Then b^* is also an ϵ -equilibrium of Γ . In other words, for every ϵ , the set of ϵ -equilibria of such a game is closed.*

To establish the claim, suppose that b^* is not an ϵ -equilibrium of Γ . Therefore for some player i and some $\delta > 0$, there exists a deviation $b_i \in B_i$ such that:

$$\Pi_i(b_i, b_{-i}^*) - \Pi_i(b^*) \geq \epsilon + 3\delta$$

By continuity of Π , we have for a large enough k :

$$\begin{aligned}\Pi_i(b^k) - \Pi_i(b^*) &\leq \delta \\ \Pi_i(b_i, b_{-i}^*) - \Pi_i(b_i, b_{-i}^k) &\leq \delta\end{aligned}$$

Subtracting the two above inequalities to the previous one yields:

$$\Pi_i(b_i, b_{-i}^k) - \Pi_i(b^k) \geq \epsilon + \delta$$

which contradicts the premise that b^k is an ϵ -equilibrium. This completes the proof of Claim 3.

We have now all ingredients to prove the lemma. Assume first that b^* is an ϵ -equilibrium in Γ and define b^k to be the strategy profile induced by b^* on Γ_k and b'^k induced by b^k on Γ . By

Claim 2, for every k , b^k is a $(\epsilon + 2w^k)$ -equilibrium in Γ_k , and by Claim 1, $w^k \rightarrow 0$. Furthermore, $d(b'^k, b^*) \leq 2^{-k}$ by definition of d , therefore $b'^k \rightarrow b^*$.

Conversely, assume now that there exists such sequences ϵ^k , t^k and b^k . By Claim 2, for every k , the strategy profile b'^k induced by b^k on Γ is an $(\epsilon^k + w^{t^k})$ -equilibrium in Γ . Since Γ is continuous, we know by Claim 1 that $(\epsilon^k + w^{t^k}) \rightarrow \epsilon$. For every $\delta > 0$, we have for every k large enough $\epsilon^k + w^{t^k} \leq \epsilon + \delta$. By Claim 3, this implies that b^* is a $(\epsilon + \delta)$ -equilibrium. Since this is true for every $\delta > 0$, we have shown that b^* is an ϵ -equilibrium in Γ . \square

4.7 Truncation of Perturbed Games

The goal of this section is to extend the results of the previous section to trembling hand perfect ϵ -equilibria (instead of just for ϵ -equilibria). We define what is the truncation of a perturbed game in a straightforward way: for every perturbed game $\hat{\Gamma} = (\Gamma, \eta)$ and every positive integer t , the *truncated perturbed game* $\hat{\Gamma}_t$ is $(\Gamma_t, \tilde{\eta})$, where $\tilde{\eta}$ is simply the restriction of η to X_t .

Lemma 6 *In a continuous game Γ , a strategy profile b^* is a trembling hand perfect ϵ -equilibrium if and only if there exists sequences ϵ^k , t^k and b^k such that for every k , b^k is a trembling hand perfect ϵ^k -equilibrium in Γ_{t^k} , and, as k goes to infinity, $\epsilon^k \rightarrow \epsilon$, $t^k \rightarrow \infty$ and $b^k \rightarrow b^*$, where b^k is induced by b^k on Γ .*

Proof. We first show the following.

Claim 4 *Let Γ be a continuous game and Γ_t its truncation.*

1. *Any trembling hand perfect ϵ -equilibrium \tilde{b}^* in Γ_t induces a trembling hand perfect $(\epsilon + w^t)$ -equilibrium in Γ .*
2. *Any trembling hand perfect ϵ -equilibrium b^* in Γ induces a trembling hand perfect $(\epsilon + 2w^t)$ -equilibrium in Γ_t .*

For proving the first item of Claim 4, let \tilde{b}^* be a trembling hand perfect ϵ -equilibrium in Γ_t . By definition of a trembling hand perfect equilibrium, there exists \tilde{b}^k a sequence of ϵ -equilibrium of a test sequence $\hat{\Gamma}_t^k = (\Gamma_t, \tilde{\eta}^k)$ of the game Γ_t , such that $\tilde{b}^k \rightarrow \tilde{b}^*$. This test sequence can be extended into a sequence of infinite games $\hat{\Gamma}^k = (\Gamma, \eta^k)$ by choosing the same minimal probabilities for any information set of round at most t , and for every information set u of round greater than t and any action $a \in A(u)$, define $\eta_u^k(a) = \frac{2^{-k}}{|A(u)|}$. Therefore the sequence $\hat{\Gamma}^k$ is by definition a test sequence of Γ . Furthermore, η^k was defined such that Claim 2 applies: the strategy profiles b^k induced by \tilde{b}^k are therefore $(\epsilon + w^t)$ -equilibria of $\hat{\Gamma}_t^k$. We also have that b^k is induced by \tilde{b}^k , b^* is induced by \tilde{b}^* and $\tilde{b}^k \rightarrow \tilde{b}^*$, therefore $b^k \rightarrow b^*$. This proves that b^* is a trembling hand perfect $(\epsilon + w^t)$ -equilibrium of Γ .

For proving the second item of Claim 4, let b^* be a trembling hand perfect ϵ -equilibrium in Γ . By definition, there exists a sequence b^k of ϵ -equilibria of a test sequence $\hat{\Gamma}^k$ of Γ such that $b^k \rightarrow b^*$. For k large enough, the following is true:

$$\forall u \in U, \forall a \in A(u), \quad \eta_u^k(a) \leq \frac{1}{|A(u)|}$$

Therefore we can apply Claim 2: the strategy profiles \tilde{b}^k induced by b^k in $\hat{\Gamma}_t^k$ are $(\epsilon + 2w^t)$ -equilibria. Obviously $\hat{\Gamma}_t^k$ is a test sequence for Γ_t . And since \tilde{b}^k is induced by b^k , \tilde{b}^* is induced by b^* and $b^k \rightarrow b^*$, we know that \tilde{b}^k converges to \tilde{b}^* . Which proves that \tilde{b}^* is a $(\epsilon + 2w^t)$ -equilibrium in Γ_t . This completes the proof of Claim 4.

Claim 5 *Let b^k be a sequence of trembling hand perfect ϵ -equilibria of a game Γ such that $b^k \rightarrow b^*$. Then b^* is also a trembling hand perfect ϵ -equilibrium of Γ .*

Indeed, for every k , since b^k is a trembling hand perfect ϵ -equilibrium, there exists a sequence $b^{k,n}$ of ϵ -equilibria of a test sequence $\hat{\Gamma}^n$ of Γ , such that $b^{k,n} \xrightarrow{n \rightarrow \infty} b^k$. The sequence $b^{k,k}$ of ϵ -equilibria of the test sequence $\hat{\Gamma}^k$ converges to b^* , thus proving that it is a trembling hand perfect ϵ -equilibrium of Γ .

The proof of the lemma is then identical to the proof of Lemma 5, by replacing Claims 2 and 3 with Claims 4 and 5, respectively. \square

4.8 Proof of Lemma 1

We have now all the ingredients to prove that, as stated in Lemma 1 every infinite, continuous, measurable, well-rounded, extensive game with perfect recall and finite action set has a trembling hand perfect equilibrium. First observe that the set of strategy profiles B of an extensive game Γ with a finite action set is sequentially compact. Indeed, let Γ be a game with a finite action set A . For every player i and every information set u , the set of local strategies B_{iu} is a simplex in the space $\mathbb{R}^{A(u)}$, and therefore it is sequentially compact. Because a countable product of sequentially compact spaces is sequentially compact,

$$B = \prod_{i \in N} \prod_{u \in U_i} B_{iu}$$

is sequentially compact. Indeed, there are countably many finite histories in any game with a finite action set, therefore there are also countably many information sets.

Consider now a (possibly infinite) continuous well-rounded extensive game Γ with perfect recall and a finite action set, and consider the corresponding sequence of truncated games Γ_k . Selten [30] has shown in 1975 that every finite extensive game with perfect recall has a trembling hand perfect equilibrium. By this result, every game Γ_k has a trembling hand perfect equilibrium that we call b^k . We call b'^k the sequence of strategy profiles induced by b^k in the game Γ . Because B is sequentially compact, we can extract a convergent subsequence b'^{t_k} and we call b^* its limit. Lemma 6 shows that b^* is a trembling hand perfect equilibrium of Γ .

We say that Γ is a *symmetric game* if it satisfies the following requirements. (1) For every player i , there exist a bijection $type_i : U_i \rightarrow [1..I]$ such that for every two players i and i' and two information sets $u \in U_i$ and $u' \in U_{i'}$ such that $type_i(u) = type_{i'}(u')$, we also have $A(u) = A(u')$ and $actions_i(u) = actions_{i'}(u')$. (2) For every two players i and i' , we define the relation $R_{i,i'}$ on strategy profiles: For every two strategy profiles b and b' , we have $b R_{i,i'} b'$ if and only if, for every positive integer k ,

$$b_{i,type_i^{-1}(k)} = b'_{i',type_{i'}^{-1}(k)} \text{ and } b'_{i,type_i^{-1}(k)} = b_{i',type_{i'}^{-1}(k)},$$

and, for every player j different from i and i' , $b_j = b'_j$. To be symmetric, Γ must satisfy that, for every two players i and i' and every two strategy profiles b and b' such that $b R_{i,i'} b'$, we have

$\Pi_i(b) = \Pi_{i'}(b')$ and $\Pi_i(b') = \Pi_{i'}(b)$. In a symmetric game Γ , we say that a profile of strategies $b \in B$ is *symmetric* if and only if for every two players i and i' , $b R_{i,i'} b$. Since a limit of a sequence of symmetric strategy profiles is symmetric, we can derive that b^* is a symmetric trembling-hand perfect equilibrium. \square

5 Existence of efficient robust algorithms for LCL games

The hypotheses regarding the topological nature of the strategy, and on the nature of the payoff function (continuity, measurability, etc.) are standard in the framework of extensive games. The notion of well-rounded game is new, and used to capture the fact that the nodes play in rounds in an LCL game. The fact that the equilibrium is symmetric is crucial since, in LCL games, as in randomized distributed computing in general, the instructions given to all nodes are identical, and the behavior of the nodes only vary along with the execution of the algorithm when they progressively discover their environment. We show that LCL games satisfy all hypotheses of Lemma 1, from which we get our main result as stated in the introduction:

Theorem 1 *Let \mathcal{L} be a greedily constructible locally checkable labeling. The LCL game associated to \mathcal{L} has a symmetric trembling-hand perfect equilibrium.*

The rest of the section is dedicated to the proof of the theorem. We start by formally defining LCL games.

5.1 Formal Definition of a LCL Game

Let A be a finite alphabet, \mathcal{F} a family of graphs with at most n vertices, \mathbf{D} a probability distribution over \mathcal{F} , and \mathcal{L} a greedily constructible LCL language over \mathcal{F} . Let t be the radius of \mathcal{L} and $\text{good}(\mathcal{L})$ be the set of good balls in \mathcal{L} . Let $\text{pref} : \text{good}(\mathcal{L}) \mapsto [0, M]$, for $M > 0$, be a function representing the preferences of the players over good balls and $\delta \in (0, 1)$ a *discounting factor*. We define the game

$$\Gamma(\mathcal{L}, \mathbf{D}, \text{pref}, \delta) = (N, A, X, P, U, p, \pi)$$

associated to the language \mathcal{L} , the distribution \mathbf{D} , the preference function pref , and the discounting factor δ , as follows.

- The player set is $N = \{1, \dots, n\}$.
- The action set is $A \cup \mathcal{F}$ where the actions in \mathcal{F} are only used by the extra player Chance in the initial move, and the actions in A are used by the actual players.
- The first move of the game is made by Chance (i.e $P(\emptyset) = c$). As a result, a graph $G \in \mathcal{F}$ with a mapping of the players to the nodes of G is selected at random according to the probability distribution \mathbf{D} . From now on, the players are identified with the vertices of the graph G , labeled from 1 to n . Note that \mathcal{F} might be reduced to a single graph, e.g., $\mathcal{F} = \{C_n\}$, and Chance just selects which vertices of the graph will be played by which players.
- The game is then divided into rounds (corresponding to the intuitive meaning in synchronous distributed algorithms). At each round, the *active* players play in increasing order, from 1 to n . At round 0 every player is active and plays, and every action in A is available.

- At the end of each round (i.e., after every active player has played the same number of times), some players might become *inactive*, depending on the actions chosen during the previous rounds. For every $i \in N$, let $s(i)$ denote the last action played by player i , which we call the *state* of i , and let $ball(i)$ denote the ball of radius t centered at node i . Every player i such that $ball(i) \in good(\mathcal{L})$ at the end of a round becomes *inactive*.
- In subsequent rounds, the set of available actions might be restricted. For every round $r > 0$, and for every active player i , an action $a \in A$ is available to player i if and only if there exists a ball $b \in good(\mathcal{L})$ compatible with the states of inactive players in which $s(i) = a$.
- A history is terminal if and only if either it is infinite or it comes after the end of a round and every player is inactive after that round.
- Let x be a history. We denote by $actions_i(x)$ the sequence of actions extracted from x by selecting all actions taken by player i during rounds before $r(x)$. (The action possibly made by player i at round $r(x)$, and actions made by a player $j \neq i$ are not included in $actions_i(x)$).
- Let x and y be two non terminal histories such that $P(x) = P(y) = i$. Then x and y are in the same information set if and only if for every $j \in ball(i)$, we have

$$actions_j(x) = actions_j(y).$$

This can be interpreted by saying that a player i “knows” every action previously taken by any player at distance at most t .

- Let i be a player, and let z be a terminal history. We define the *terminating time* $time_i(z)$ of player i in history z by

$$time_i(z) = \max\{|actions_j(z)|, j \in ball(i)\} - 1.$$

The payoff function π of the game is then defined as follows. For every player i , and every terminal history z ,

$$\pi_i(z) = \delta^{time_i(z)} \cdot pref(ball(i))$$

And $\pi_i(z) = 0$ if $time_i(z) = \infty$.

5.2 Properties of LCL Games

Fact 5 *LCL games are well-rounded.*

Proof. This follows directly from the fact that, in a LCL game, every active player plays at every round until it becomes inactive, and, once inactive, a player cannot become active again. \square

Fact 6 *LCL games have perfect recall.*

Proof. Let $\Gamma(\mathcal{L}, \mathbf{D}, pref, \delta) = (N, A, X, P, U, p, \pi)$ be an LCL game. Let u and u' be two information sets of the same player i , for which there exists $x \in u$, $x' \in u'$ and $a \in A(u')$, such that $(x', a) \preceq x$. Let y be a history in u . Since x and y are in the same information set u , it follows that, for every player $j \in ball(i)$, we have $actions_j(x) = actions_j(y)$. In particular, this implies that x and y are in

the same round. Let y' be the only history which is a prefix of y with $P(y') = i$ and $r(y') = r(x')$. (Such a history exists because $r(x') < r(x)$ and $r(y) = r(x)$). Since the players play in the same order at every round, we get that, for every player $j \in \text{ball}(i)$, $\text{actions}_j(x') = \text{actions}_j(y')$. Therefore $y' \in u'$. Furthermore, since $\text{actions}_i(x) = \text{actions}_i(y)$, the action played by i after y' must be a , which implies $(y', a) \preceq y$, and concludes the proof. \square

Fact 7 *The payoff function π of a LCL game is measurable on the σ -algebra Σ corresponding to the game.*

Proof. We prove that, for every player i and every $a \in \mathbb{R}$, $\pi_i^{-1}(]a, +\infty[) \in \Sigma$, which implies that π is measurable on Σ . Without loss of generality, let us assume that $M = 1$, that is, $\pi_i : Z \mapsto [0, 1]$. For every $a < 0$, we have $\pi_i^{-1}(]a, +\infty[) = Z$, and thus is in Σ . Similarly, for every $a > M$, we have $\pi_i^{-1}(]a, +\infty[) = \emptyset \in \Sigma$. So, let us assume that $a \in]0, 1]$ and let z be a terminal history such that $\pi_i(z) > a$, which implies that $\text{time}_i(z) < \frac{\ln a}{\ln \delta}$, i.e every player in $\text{ball}(i)$ has played only a finite number of times in the history z . Let x be the longest history such that $x \preceq z$ and $r(x) = \text{time}_i(z)$. Then the history x' that comes just after x in z is the shortest prefix of z such that every player in $\text{ball}(i)$ is now inactive. Let z' be a terminal history such that $x' \preceq z'$. Since every player $j \in \text{ball}(i)$ is inactive after x' , it follows that the state of any such player in z' is the same as its state in z . Therefore $\pi_i(z') = \pi_i(z)$. It follows from the above that, for any terminal history z such that $\pi_i(z) > a$, there exists a finite history x' in round $\text{time}_i(z) + 1$ such that $z \in Z_{x'} \subseteq \pi_i^{-1}(]a, +\infty[)$. Since there can only be a finite number of histories in round $\text{time}_i(z)$, we get that $\pi_i^{-1}(]a, +\infty[)$ is the union of a finite number of sets of the form $Z_{x'}$. As a consequence, it is measurable in Σ . It remains to prove that $\pi_i^{-1}(]0, +\infty[) \in \Sigma$. actually, this simply follows from the fact that $\pi_i^{-1}(]0, +\infty[) = \bigcup_{k \geq 1} \pi_i^{-1}(] \frac{1}{k}, +\infty[)$, and the fact that Σ is stable by countable unions. \square

Fact 8 *LCL games are continuous.*

Proof. Let b be a strategy profile, and let $(b^k)_{k \geq 0}$ be a sequence of strategy profiles such that $d(b^k, b) \rightarrow 0$ when $k \rightarrow \infty$. By definition of the metric d on B (cf. subsection 4.4), we have that $d(\rho_{b^k}, \rho_b) \rightarrow 0$ when $k \rightarrow \infty$. By definition of the metric on O , we have that, for any finite history x ,

$$|\rho_{b^k}(x) - \rho_b(x)| \xrightarrow[k \rightarrow \infty]{} 0.$$

It follows that for any set of the form Z_x as defined in subsection 4.2,

$$|\mu_{b^k}(Z_x) - \mu_b(Z_x)| \xrightarrow[k \rightarrow \infty]{} 0.$$

In other words the sequence of measures μ_{b^k} *strongly converges* to μ_b . Since, for every player i , the function π_i is measurable and bounded, it follows that

$$\int_{\Sigma} \pi_i d\mu_{b^k} \xrightarrow[k \rightarrow \infty]{} \int_{\Sigma} \pi_i d\mu_b.$$

Therefore, $\Pi_i(b^k) \xrightarrow[k \rightarrow \infty]{} \Pi_i(b)$, and thus the expected payoff function Π is continuous. \square

5.3 Proof of Theorem 1

The above claims show that every LCL game satisfies the requirements of Lemma 1, and therefore has a symmetric trembling-hand perfect equilibrium. \square

6 Conclusion

The objective of this paper is to address the issue of selfish behaviors in the context of distributed network computing. Theorem 1 establishes that distributed algorithms coping with selfishness do exist for a large fraction of locally checkable labeling tasks. This result is the necessary first step toward designing distributed algorithms in which the players have no incentive to deviate from their given instructions. One direction for further research consists in looking for efficient (centralized) algorithms for computing the equilibria, as well as (distributed) protocols leading the nodes to automatically adopt the desired behavior, i.e., the one with highest social benefit, either in term of converging time or quality of the solution, potentially using incentives. Another question for further studies is to measure the efficiency of trembling-hand Nash equilibria for LCL games in term of round-complexity. Apart from a few special cases, like, e.g., complete networks, these issues are challenging problems, but quite rewarding both theoretically and practically. Finally, from a broader perspective, it would be desirable to characterize efficient distributed algorithms in terms of a suitable notion of equilibrium in (extensive form) games.

References

- [1] Noga Alon, László Babai, and Alon Itai. A fast and simple randomized parallel algorithm for the maximal independent set problem. *J. Algorithms*, 7(4):567–583, 1986.
- [2] David Auger, Johanne Cohen, Pierre Coucheney, and Lise Rodier. Distributed selfish algorithms for the max-cut game. In *28th Int. Symposium on Computer and Information Sciences (ISCIS)*, pages 45–54, 2013.
- [3] Leonid Barenboim and Michael Elkin. *Distributed Graph Coloring: Fundamentals and Recent Developments*. Synthesis Lectures on Distributed Computing Theory. Morgan & Claypool Publishers, 2013.
- [4] Subir K Chakrabarti. Equilibrium in Behavior Strategies in Infinite Extensive Form Games with Imperfect Information. *Economic Theory*, 2(4):481–494, 1992.
- [5] Yi-Jun Chang, Tsvi Kopelowitz, and Seth Pettie. An exponential separation between randomized and deterministic complexity in the LOCAL model. In *57th IEEE Symposium on Foundations of Computer (FOCS)*, 2016.
- [6] Raphael Eidenbenz and Roger Wattenhofer. Good programming in transactional memory: Game theory meets multicore architecture. *Theor. Comput. Sci.*, 412(32):4136–4150, 2011.
- [7] Ky Fan. Fixed-Point and Minimax Theorems in Locally Convex Topological Linear Spaces. *PNAS*, 38(2):121–126, 1952.
- [8] Laurent Feuilloley and Pierre Fraigniaud. Randomized local network computing. In *27th ACM Symposium on Parallelism in Algorithms and Architectures (SPAA)*, pages 340–349, 2015.
- [9] Pierre Fraigniaud, Marc Heinrich, and Adrian Kosowski. Local conflict coloring. In *57th IEEE Symposium on Foundations of Computer Science (FOCS)*, 2016.
- [10] Pierre Fraigniaud, Amos Korman, and David Peleg. Towards a complexity theory for local distributed computing. *J. ACM*, 60(5):35, 2013.
- [11] Drew Fudenberg and David Levine. Subgame-Perfect Equilibria of Finite- and Infinite-Horizon Games. *Journal of Economic Theory*, 31(2):251–268, 1983.
- [12] Mohsen Ghaffari. An improved distributed algorithm for maximal independent set. In *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2016.
- [13] Irving L. Glicksberg. A Further Generalization of the Kakutani Fixed Point Theorem with Application to Nash Equilibrium Points. *Proceedings of the AMS*, 3(1):170–174, 1952.
- [14] Christopher Harris. Existence and Characterization of Perfect Equilibrium in Games of Perfect Information. *Econometrica: Journal of the Econometric Society*, pages 613–628, 1985.
- [15] David G. Harris, Johannes Schneider, and Hsin-Hao Su. Distributed $(\Delta + 1)$ -coloring in sublogarithmic rounds. In *48th ACM Symposium on the Theory of Computing (STOC)*, 2016.

- [16] Fabian Kuhn, Thomas Moscibroda, and Roger Wattenhofer. Radio network clustering from scratch. In *12th European Symposium on Algorithms (ESA)*, LNCS 3221, pages 460–471. Springer, 2004.
- [17] Harold W. Kuhn. Extensive Games. In *Proceedings of the National Academy of Sciences of the United States of America*, volume 36, pages 570–576, 1950.
- [18] Harold W. Kuhn. Extensive Games and the Problem of Information. In *Contributions to the Theory of Games*, volume II (Annals of Mathematics Studies, 28), pages 193–216. Princeton University Press, 1953.
- [19] Michael Luby. A simple parallel algorithm for the maximal independent set problem. *SIAM J. Comput.*, 15(4):1036–1053, 1986.
- [20] Thomas Moscibroda and Roger Wattenhofer. Maximal independent sets in radio networks. In *24th ACM Symposium on Principles of Distributed Computing (PODC)*, pages 148–157, 2005.
- [21] Moni Naor and Larry J. Stockmeyer. What can be computed locally? *SIAM J. Comput.*, 24(6):1259–1277, 1995.
- [22] John F. Nash. Equilibrium Points in n-Person Games. *Proceedings of the National Academy of Sciences*, 36(1):48–49, 1950.
- [23] Noam Nisan. Algorithms for selfish agents. In *16th Symposium on Theoretical Aspects of Computer Science (STACS)*, pages 1–15, 1999.
- [24] Noam Nisan and Amir Ronen. Algorithmic mechanism design. In *31st ACM Symposium on Theory of Computing (STOC)*, pages 129–140, 1999.
- [25] Noam Nisan, Michael Schapira, Gregory Valiant, and Aviv Zohar. Best-response mechanisms. In *Innovations in Computer Science (ICS)*, pages 155–165, 2011.
- [26] Bezalel Peleg. Equilibrium Points for Games with Infinitely Many Players. *Journal of the London Mathematical Society*, 1(1):292–294, 1969.
- [27] David Peleg. *Distributed Computing: A Locality-Sensitive Approach*. SIAM, 2000.
- [28] Roy Radner. Collusive Behavior in non-Cooperative epsilon-Equilibria of Oligopolies with Long but Finite Lives. *Journal of Economic Theory*, 22:136–154, 1980.
- [29] Reinhard Selten. Spieltheoretische Behandlung eines Oligopolmodells mit Nachfragetraheit. *Z. Gesamte Staatswissenschaft*, 12:301–324, 1965.
- [30] Reinhard Selten. Reexamination of the Perfectness Concept for Equilibrium Points in Extensive Games. *International journal of game theory*, 4(1):25–55, 1975.
- [31] Jukka Suomela. Survey of local algorithms. *ACM Comput. Surv.*, 45(2):24, 2013.